VERITAS

NetBackup Vault Extension
Functional Design
Revision 3.4

F. Bryan Cooper

VAULT MANAGEMENT

Overview

This Functional Design document provides an architectural services-oriented approach to building and maintaining a client/server based Vault Management system. This document defines the functional requirements, develops an architecturally-integrated set of services to meet those requirements, and documents the functional capacity for each service. To provide additional technical detail, this functional design further defines the technical components and technical design needed to provide these services. A final section lists operational procedures needed to deliver each service and assigns basic levels of staff responsibility.

The chief benefit of this functional design is the consistent information for business, technical, and operational viewpoints. High level architectural service design provides a business understanding and real-world value. Technical implementation considerations are shown in the technical component diagrams and definitions. Procedural charts provides a hands-on understanding for operational staff and clarifies areas of responsibility.

1) **Architectural Services**: A set of interrelated services is designed to meet the overall goals and/or challenges. This service model provides a framework for all functional features, and creates a visual model for further technical discussion. The services section uses tables and diagrams to create an organized and expandable system for future enhancements.
2) **Technical Components**: This -section ties together the conceptual services with the actual NetBackup software and/or other software as needed. These diagrams and tables show how services are implemented and fundamentally interrelate. Tools definitions provide a technical design summarization for architectural considerations, and includes basic interface specifications.
3) **Operational Procedures**: This final sub-section shows how the tools are to be used. The procedural diagrams and tables provide a real-world understanding, externalizing any assumptions about who would use the Tools and in what manner.

Other Related Services

All Storage Management services are interdependent in some degree or another. The Vault Management System is reliant upon both Backup and Media Management services. It is designed to integrate with other Operational Services sub-systems, such as Event Management and Help Desk. The interdependency on these other, operational services creates the functional need for close integration of Storage Management services. Different Functional Designs provide detailed information for the different Storage Management sub-systems. It is beyond the scope of this Functional Design to document these other service areas, such as Help Desk. However, their value to an production environment is noted to provide an operational context.

In this document, references to NetBackup's Volume Manager will be as "Media Manager". NetBackup's Volume Manager will be renamed in a future release to avoid conflict with VERITAS Volume Manager.

Other Related Vault Extension Documents:

A **System Administration Guide** provides basic installation, configuration and troubleshooting. The **Operations Guide** provides day to day procedures to follow to work with vault reports, tapes and vault vendors. **Code Design** provides internal information on how bpvault is written.
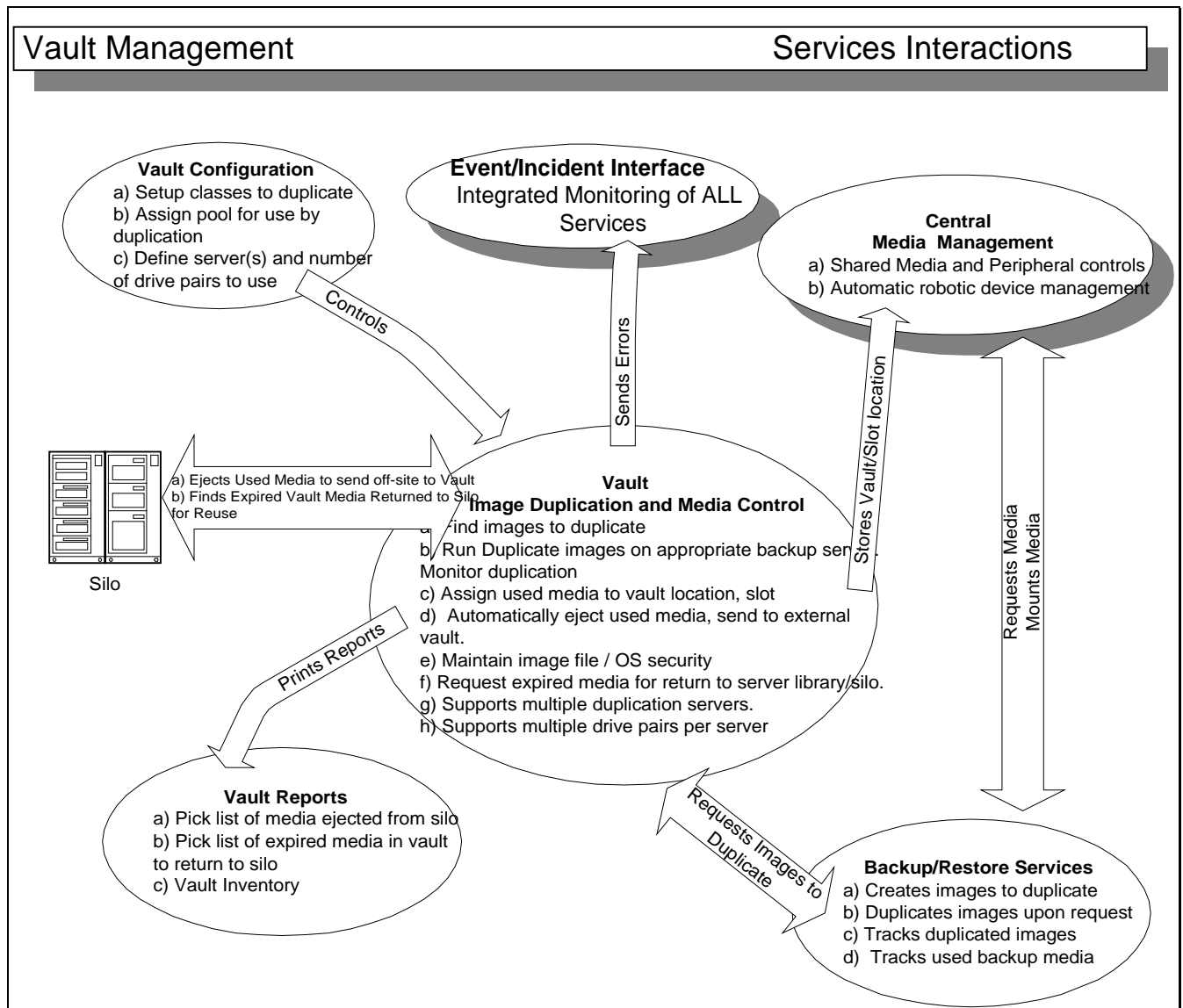
Architectural Services

A Vault Management system is a server-centric approach that provides both backup duplication and off-site storage and retrieval of media. A Vault system duplicates backup images onto tape or other media and simplifies restoring the duplicated files when the original backup image media is damaged or unavailable. A master-slave implementation extends the storage to other machines by centrally controlling duplication for multiple backup servers simultaneously. A short list of basic services includes:

1.   Provides additional backup protection by duplicating backup images.
2.   Optional vaulting of original images/tapes.
3.   Can directly support different media types for both backup and duplication media.
4.   Can backup images onto different media types to support optimal cost-effective configurations.
5.   Maintains file OS level information and security.
6.   Implementation must be scaleable, distributed and heterogeneous in design.
7.   Controls tape storage within tape silo vs. tapes ejected from silo for transference to off-site vault.
8.   Assigns vault slot id location when required for use by vault vendor.
9.   Prints appropriate reports
10.   Uses existing NetBackup capabilities for key features to ensure compatibility and robustness:
   a)   Maintains duplication image catalog for Recovery services.
   b)   Uses existing Media Manager services for fundamental media and robotic management and control.
   c)   Uses existing Media Manager database for keeping track of media containing duplicated images.
   d)   Uses NetBackup backup image catalog to keep track of which images need to be duplicated and which images are already duplicated.
   e)   Uses NetBackup media catalog to determine expiration data of used media.

The overall Architectural design is defined as a distributed client/server system. Client/server systems provide several basic features:

1)   Built-in network support. All services expect to exist within a network topology and thus do not require special configuration or software such as found in older, standalone designs.
2)   Scaleable support for large numbers of backup servers. Server based features support faster processors and faster devices.
3)   Peer to peer controls. Servers can control other servers to provide better load balancing and different network topologies and bandwidths. Distributed servers ensure better production support and redundancy.
4)   Remote installation, control and configuration. Centralized management reduces management costs by making it easier to setup and run basic backup operations.
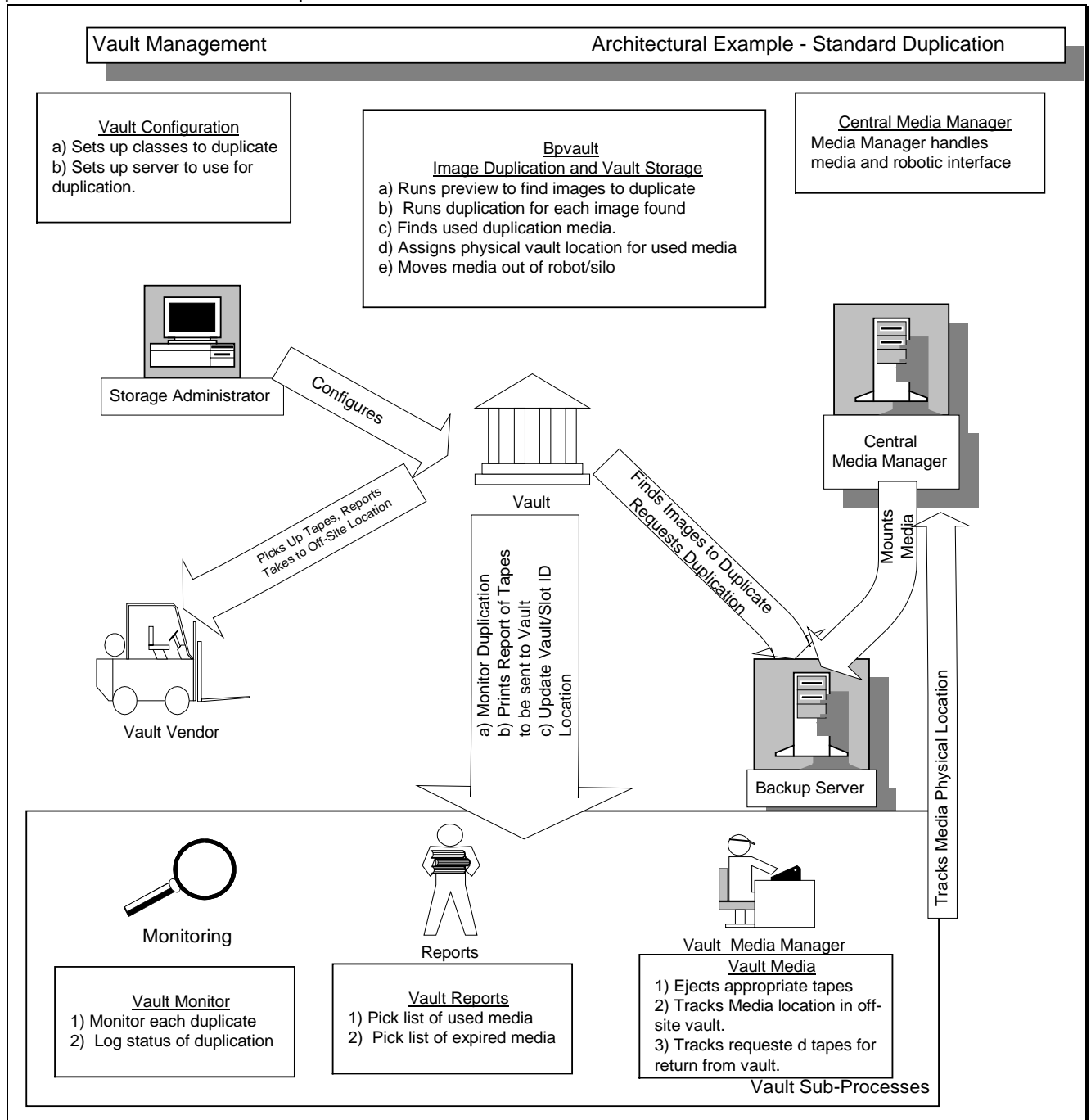
This diagram provides an Architectural overview of the basic client/server backup design:

Vault Management   Services Interactions

Vault Configuration
a) Setup classes to duplicate
b) Assign pool for use by
duplication
c) Define server(s) and number
of drive pairs to use

Event/Incident Interface
Integrated Monitoring of ALL
Services

**Central
Media Management**
a) Shared Media and Peripheral controls
b) Automatic robotic device management

Controls

Sends Errors

Stores Vault/Slot location

Silo

a) Ejects Used Media to send off-site to Vault
b) Finds Expired Vault Media Returned to Silo
for Reuse

**Vault
Image Duplication and Media Control**
a) Find images to duplicate
b) Run Duplicate images on appropriate backup server,
Monitor duplication
c) Assign used media to vault location, slot
d) Automatically eject used media, send to external
vault.
e) Maintain image file / OS security
f) Request expired media for return to server library/silo.
g) Supports multiple duplication servers.
h) Supports multiple drive pairs per server

Requests Media
Mounts Media

Prints Reports

Requests Images to
Duplicate

Vault Reports
a) Pick list of media ejected from silo
b) Pick list of expired media in vault
to return to silo
c) Vault Inventory

Backup/Restore Services
a) Creates images to duplicate
b) Duplicates images upon request
c) Tracks duplicated images
d) Tracks used backup media

The overall set of client/server Architectural Services is provided in this chart:

Hierarchical Storage Management		Architectural Services
Service	Business Challenge	Functional Capability
Vault Duplication	Protect Information - make sure appropriate images are duplicated Enterprise Scalability - support multiple servers	a) Determine backup images to duplicate. b) Duplicate images on multiple drives, multiple servers. c) Duplicate during day without using production network bandwidth. d) Optional vaulting of original images without duplication requirement. e) Can duplication locally, across LAN or WAN.
Vault monitoring	Fast response	a) Monitor duplication process for successful completion b) Support interface to Event Management
Vault Configuration		a) Setup use of servers, which images to duplicate, other options.
Vault Reports	Protect Information - make sure media location is known	a) Print appropriate reports for sending media off-site to vault and returned from vault. b) Print regular inventory of vault.
Vault Media Management	Reduce Costs - reduce manual administration of media	a) Keep track of used media location in vault. b) Eject used media from silo for duplication session. c) Find returned, expired media and re-use in silo
Backup / Restore Services	Protect Information -	a) Creates duplicate on image by image basic. b) De-multiplexes backup image during duplication. c) Keeps track of both backup images and duplicated copy d) Keeps track of media used by duplicated copy. e) Simple image catalog change to restore from copy.
Media Manager interface	Protect Information	a) Maintains media used information b) Maintains vault location information.

This chart shows an example of how these services actually work. Detailed technical examples are provided in the Technical Components section of this document:

Vault Management   **Architectural Example - Standard Duplication**

Vault Configuration
a) Sets up classes to duplicate
b) Sets up server to use for duplication.

Bpvault
Image Duplication and Vault Storage
a) Runs preview to find images to duplicate
b) Runs duplication for each image found
c) Finds used duplication media.
d) Assigns physical vault location for used media
e) Moves media out of robot/silo

Central Media Manager
Media Manager handles media and robotic interface

Storage Administrator

Configures

Picks Up Tapes, Reports
Takes to Off-Site Location

Vault

Finds Images to Duplicate
Requests Duplication

Central
Media Manager

Mounts
Media

a) Monitor Duplication
b) Prints Report of Tapes
to be sent to Vault
c) Update Vault/Slot ID
Location

Vault Vendor

Backup Server

Tracks Media Physical Location

Monitoring

Reports

Vault Media Manager
Vault Media
1) Ejects appropriate tapes
2) Tracks Media location in off-site vault.
3) Tracks requeste d tapes for return from vault.

Vault Monitor
1) Monitor each duplicate
2) Log status of duplication

Vault Reports
1) Pick list of used media
2) Pick list of expired media

Vault Sub-Processes

Technical Components

The functional requirements outlined in the architectural services section provides us with a general understanding of the Vault Management capabilities. Additional implementation specific issues are critical to provide the best quality features. For example, functional scalability creates several technical issues: network bandwidth, catalog sizing, administration, etc. In this section, we list the specific components which deliver the architecture, and review how each component is designed to overcome various technical issues.

The various components for Vault Management includes:

1) Vault Batch Processing
2) Vault Duplication
3) Vault Duplication monitoring
4) Vault Configuration
5) Vault Report
6) Vault Media Management
7) Backup Image Duplication
8) Existing NetBackup Services
9) Media Manager interface

A short list of the technical design issues includes:

1) Initiate and control duplications from a centralized location
2) Initiate and control restores of duplicated image from a centralized location
3) Control various duplication parameters by use of NetBackup classes.
4) Reduce duplicating data over the network by duplicating local whenever possible.
5) Support automated retry of duplication.
6) Soft shutdown of duplication.
7) Direct appropriate duplications to either master or slave servers.
8) Support one or more pairs of drives per server.
9) Duplicates can be written to a variety of storage devices and media.
10) Duplicates can span multi-volume media, yet support industry standard "tar" format for disaster recovery.
11) Duplicates to de-multiplex NetBackup images.
12) Interact with Media Management Service for media availability and media mount/unmount.
13) Duplicate restores work the same as for normal Backup/Restore, e.g. allowed to same or different client, same or different location.
14) Allow duplications to notify external operations, e.g. integrating with the *Event Management Service*.
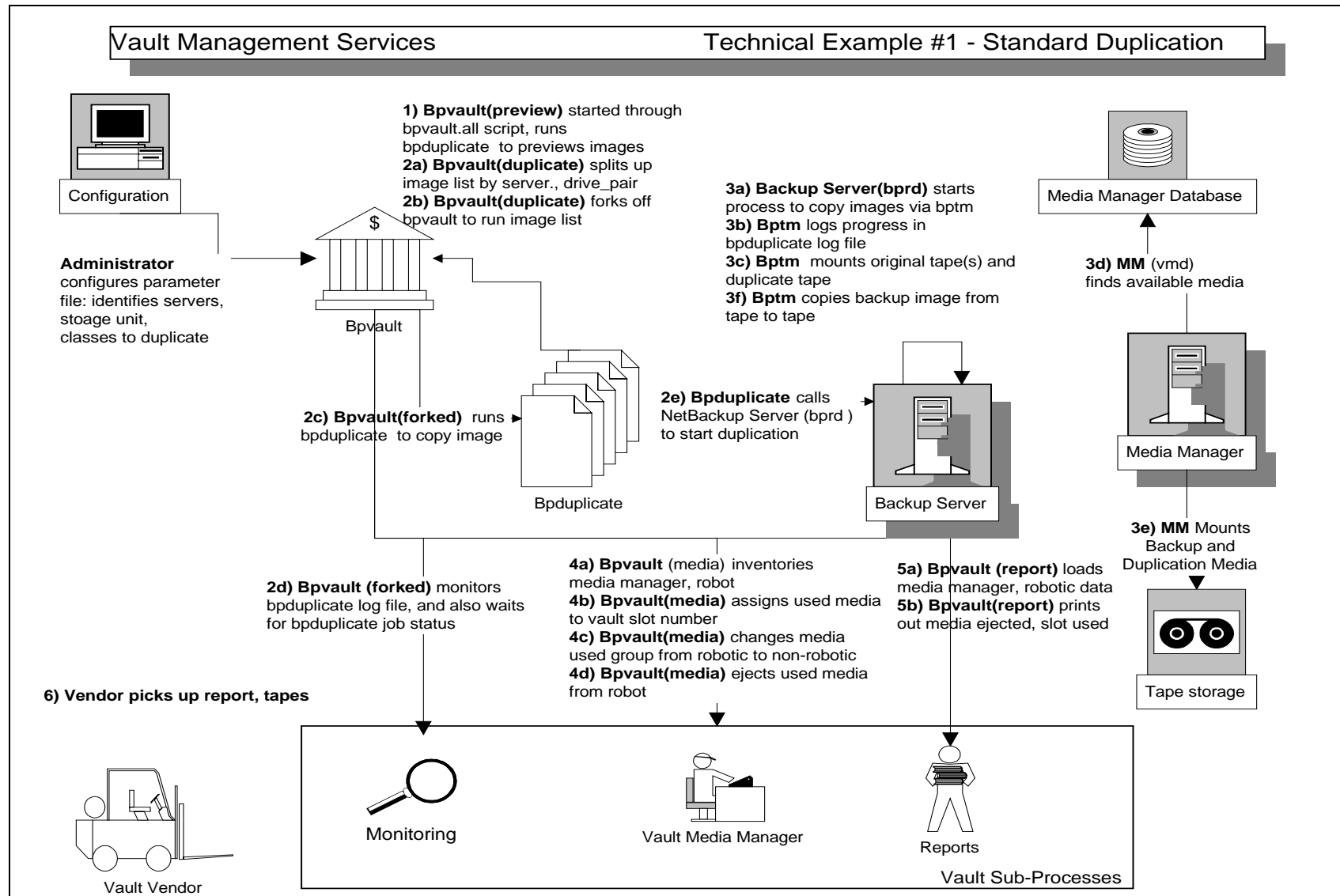15) All normal Backup/restore functions supported - e.g. data types, client types

Vault Management Technical Components			
Service	**Components**	**Functional Capability**	**Technical Design**
Vault Batch	bpvault.all, bpvault.acs, bpvault.tld	a) Organize various steps into daily/weekly batch	a) Runs bpvault utilities for each specific step. b) Simple script logic c) Uses one script for ACLS processing, another for TLD processing.
Vault Duplication	bpvault - function preview and function duplicate	a) Find all images to duplicate only for classes configured. b) Run one or more duplicates simultaneously on one or more servers	a) Run bpduplicate function Preview mode for one or more classes. b) Load balance duplications by splitting found backup images into files, one for each server/drive pair. c) Runs multiple bpduplicates in parallel. Fork off separate bpvault program to run/monitor bpduplicate. d) Run bpduplicate for each image found on server where duplicate was made if possible. (If backup server is one of the servers used for duplication.) Otherwise, run duplication on any available server. e) Can use schedule to further restrict which images to duplicate. f) Can use duplicate_days to look back multiple days for images needing duplication. g) Can bypass duplication step to only vault original images, original tapes.
Vault Duplication Monitoring	bpvault - function duplicate (forked bpvault process)	a) Monitor duplicates	a) Monitor log file from bpduplicate for errors in background. b) Ensure each bpduplicate drive pair output is monitored separately and has unique identifier for support. c) Ensure bpduplicate has exited before allowing next bpduplicate to run on drive pair.
Vault Configuration	dup_param	a) Configures which backups to duplicate	a) ascii parameter file read by bpvault; contains: 1. server(s), 2. number of drive pairs, 3. destination storage unit, 4. classes, 5. # of previous days duplicates to find, 6. name of vault, 7. NetBackup pool for duplication 8. Netbackup robotic and non-robotic group 9. Command locations

Vault Management Technical Components			
Service	Components	Functional Capability	Technical Design
Vault Report	bpvault - function report	a) Reports for picking used tapes from silo/library, for returning expired tapes from vault, for full vault inventory	a) Reports use data files created by bpvault Media Management commands b) Report "commands" are picking_library, dist_vault, dist_library, picking_vault, vault. (See SysAdmin Guide)
Vault Media Management	bpvault - command media	a) Commands to find current Media Manager invent	a) Media function volumeinv - saves Media Manager data for duplicate pool b) Media function
Backup Image Duplication	bpduplicate	a) Find images to duplicate b) Duplicate images	a) Generates list of images to duplicate b) Sends command to bptm to copy image, location of log file. c) Runs in foreground. Waits for bptm to finish.
NetBackup Backup Server	VERITAS NetBackup Server utilities bprd bpbrm bptm bpdm	a) Master server listen for duplication requests b) d) Allow user-directed backups	a) Support network duplications. b) DuplicatesTrue Image Recovery (TIR) separately for tracking file deletion. c) Maintain file / OS security d) Support multi-tape backup images
NetBackup Master / slave Backup Server	VERITAS NetBackup Master/Slave Server bprd bpcd add_slave add_slave_on_clients	a) Master server sends duplicate request to appropriate server b) Provide central backup/duplicate image catalog for restore.	a) Redirect duplication requests to appropriate slave based upon destination storage requested for specific image. b) Allows duplicate to run off-network for tape to tape copy.
NetBackup Server side Multiplexing (MPX)	VERITAS NetBackup server utilites bpbrm bptm	a) Support copy of tape multiplexing on both local backups and across networks.	a) De-multiplexing of backup image automatic by bptm. b) Only one duplicate per drive_pair. c) Duplication speed limited to tape read/write speeds. High use of multiplexing limits copy speed to tape read + tape mount for all tapes needed for de-multiplexing assuming tape read/write speeds similar.

Vault Management Technical Components			
Service	Components	Functional Capability	Technical Design
NetBackup Backup Server Database Services	VERITAS NetBackup database daemon: bpdbm	a) Provides Image information including number of copies (normally 1) b) Stores media used information for both primary copy and duplicate copy	a) Use bpdbm for all data access requests b) bpdbm daemon process always running on master server - not on slave servers. c) bpdbm creates unique backup identifier based on client name, time of day. d) bpdbm provides image catalog, number of image copies for determining which images need duplication. e) bpdbm provides backup media catalog for determining which backup media contains backup image, and which duplication media is used for copies. f) bpdbm provides restore with duplicate copy image/media when "primary copy" is 2
NetBackup Schedules - Centralized server scheduling	VERITAS NetBackup Scheduler bpsched	a) Can be used by bpvault/bpduplicate to limit duplication of a class to only images within specific schedule	a) bpduplicate preview can be limited by schedule name. Used through bpvault command line option.
NetBackup Backup Server - Restore functions	VERITAS NetBackup server daemons bprd bpcd bpbrm bptm bpdm	Works as normal for duplicated images.	a) Duplicated images are already de-multiplexed. b) Changing primary copy to 2 automatically forces any restore of that image to use the duplicated copy. c) There are currently no GUI interface to select duplicated copy- to change requires manual intervention.
NetBackup Classes - Centralized client management	VERITAS NetBackup class utilities	Works as normal. Class name used to limit duplication to specific clients.	a) Name of class or classes stored in dup_param parameter file. b) One bpduplicate preview mode run for each class found in parameter file.
NetBackup Storage Unit - Centralized resource control	VERITAS NetBackup storage unit utilities	Works as normal. Used to determine which server will run duplicate	a) Storage unit in dup_param parameter file must match up with server drive_pair count, e.g. one "dstunit" entry for each drive_pair.

Vault Management Technical Components			
Service	**Components**	**Functional Capability**	**Technical Design**
NetBackup Image Catalog Services - centralized image information	VERITAS NetBackup image catalog utilities: bptm bpimage bpimagelist bplist bpflist bpimmedia bpfrag	a) Centrally stores backup and duplicate catalog information. b) Keep track of both backup and duplicate image information: media, fragments, size, location on tape required for recovery. c) List directories/files/fragments in image catalog. d) Tracks backup and duplicate server name for each image to support master/slave duplication	a) Master server stores all image data. Duplicates notify master server of copy status - fragments, media used, but no image information necessary.
NetBackup Media Manager - centralized media control	VERITAS NetBackup media database utilities: bptm bpmedia bpmedialist bpexpdate bplabel	a) Used to determine which media used for primary backup copy. b) Can manually expire primary copies media to force use of duplicate copy . c) Backup retention periods for primary backup image is same for duplicate image. d) Expire media after retention period will expire duplicates copies. e) Allow freeze or suspend media sent off-site to keep from expiring while in need for recovery.	a) Media used for image duplicates will have same retention period, same media type. b) Duplicate media used must be assigned in Media Manager to duplicate pool named in dup_param parameter file. c) Expired duplicate media released in Media Manager triggers recall from off-site vault. d) Duplication sends image fragment, media used information to Master server.
Backup monitoring	xbpmon - VERITAS NetBackup daemon logging;	a) Duplication not currently monitored by NetBackup monitoring tools. b) Daemon log files can be used to track status in addition to bpvault log files.	a) Bpvault can show all current daemon log information on master server - use dup_param file to name command_log to monitor. b) VERBOSE option increases information to logging subdirectories. c) Notify scripts not supported. Must modify bpvault.all for special handling.

Vault Management	Technical Components		
Service	**Components**	**Functional Capability**	**Technical Design**
Backup reports	See Backup Console bperror bpimagelist cleanstats available_media support	a) Provide administrative reports needed for capacity, utilization planning, auditing purposes, technical support	a) Can load backup information into RDBMS for better reporting. b) bpimagelist provides basic information showing both copies of an image.
Media Manager interface		a) Use Media Manager Tools to request media from specific pools; mount tapes; control robotics b) Use specific MM pool for duplication	a) Must ensure media database and VM database stay in sync. b) Update MM description field to store vault name, duplication session, slot id, date requested

Vault Management Services   Technical Example #1 - Standard Duplication

Configuration

Administrator
configures parameter
file: identifies servers,
stoage unit,
classes to duplicate

1) Bpvault(preview) started through
bpvault.all script, runs
bpduplicate to previews images
2a) Bpvault(duplicate) splits up
image list by server., drive_pair
2b) Bpvault(duplicate) forks off
bpvault to run image list

$

Bpvault

3a) Backup Server(bprd) starts
process to copy images via bptm
3b) Bptm logs progress in
bpduplicate log file
3c) Bptm mounts original tape(s) and
duplicate tape
3f) Bptm copies backup image from
tape to tape

Media Manager Database

3d) MM (vmd)
finds available media

2c) Bpvault(forked) runs
bpduplicate to copy image

Bpduplicate

2e) Bpduplicate calls
NetBackup Server (bprd)
to start duplication

Backup Server

Media Manager

3e) MM Mounts
Backup and
Duplication Media

Tape storage

2d) Bpvault (forked) monitors
bpduplicate log file, and also waits
for bpduplicate job status

4a) Bpvault (media) inventories
media manager, robot
4b) Bpvault(media) assigns used media
to vault slot number
4c) Bpvault(media) changes media
used group from robotic to non-robotic
4d) Bpvault(media) ejects used media
from robot

5a) Bpvault (report) loads
media manager, robotic data
5b) Bpvault(report) prints
out media ejected, slot used

6) Vendor picks up report, tapes

Monitoring

Vault Media Manager

Reports

Vault Sub-Processes

Vault Vendor

This table lists the various steps as shown in the Example #1 diagram. Numbers in the table correspond to numbers in the diagram.

Technical Example #1		Standard Duplication/Vault	
Service	**Components**	**Incoming Program / Data Flow**	**Outgoing Program / Data Flow**
Backup Server-File System and Raw Partition	bprd - Job request daemon.	• bprd **started by** /etc/rc2.d startup script on master server via command line. Bprd must always be running to allow any backup/restore command • bprd can be **started by** xbpadm or bpadm **Initiate Request Daemon** option • bprd **called by** bpbackup, bpduplicate, bprestore, bparchive to start jobs. Uses known socket in /etc/services **Activity log:** /usr/openv/netbackup/logs/bprd on server	1) bprd **starts** bpsched on master server via CLI to handle jobs. • For manual jobs, bprd builds file list from client (e.g. bpbackup), then bprd **starts** bpsched as normal • bprd also regularly cleans up activity logs.
Schedules - Centralized backup scheduling	Master Server Scheduler bpsched	1) Scheduler **started by** bprd on master server. • Scheduler **exits** if no jobs needed to run, or monitored client job is finished **Activity Log:** /usr/openv/netbackup/logs/bpsched on Master server	2) bpsched **calls** bpdbm to obtain class information, storage unit server to use, file list, etc. 2) bpsched **starts** bptm to do basic media check prior to starting client job. 2) bpsched **starts** bpcd via inetd on appropriate storage unit server (e.g. master or slave server). 2) bpsched **sends** bpbrm command for bpcd to run, with required options, file list, log path to bpcd on server via socket 2) bpsched **monitors** bpbrm output via stderr • bpsched **signals** other bpsched processes to ensure only one scheduler is acting as main scheduler, to eliminate accidental duplication of jobs.
See Example #1 Diagrams	bpcd: Backup job proxy	2) bpcd **started by** bpsched via inetd / known socket. Acts as job proxy for bpsched **Activity log:** /usr/openv/netbackup/logs/bpcd on server for server related commands	3a) Bpcd Starts backup/restore manager bpbrm on master or slave server.
	bpbrm - Backup/Restore Manager	3a) bpbrm **started by** bpsched on storage unit server (via bpcd) . One bpbrm started for each backup or restore operation. 3a) bpbrm stderr output and exit status monitored by bpsched 3a) bpbrm **exits** to bpsched with status 4) bpbrm Receives signal() from bptm when media ready. • bpbrm sends **email** on job completion (via bpcd) **Activity log:** /usr/openv/netbackup/logs/bpbrm on server	3b) bpbrm **starts** bptm to start backups and duplication. Waits for child exit status. 4) bpbrm **starts** bpbkar on client. Waits for childed exit status.
Backup Server	bptm: Server	2) bptm **started by** bpsched to check validity of storage	3c) Parent bptm **starts** child bptm.

Service	Components	Incoming Program / Data Flow	Outgoing Program / Data Flow
Technical Example #1 — Standard Duplication/Vault			
(cont'd)	tape manager	unit. 3b) bptm **started by** bpbrm on master and slave servers via command line. One bptm started for backup/restore when using tape or optical media. 3c) Child bptm **started by** Parent bptm. 5d) bptm **messages** bpbrm once media is mounted via signal() • bptm stderr output **monitored** by bpbrm • Child bptm **exits** to parent bptm on client completion. • Parent bptm **exits** to bpbrm on server completion. **Activity log:** /usr/openv/netbackup/logs/bptm on server	5a) Parent bptm **calls** vmd on master server via known socket to find appropriate media. 5a) Parent bptm also **calls** bpdbm to compare Media Manager database with own, NetBackup Media database. 5d) Parent bptm **calls** ltid on storage unit server via known socket to mount media. 6) Child bptm **receives** data from bpbkar on client 7a) Child bptm **writes** data to buffer. 7b) Parent bptm **writes** buffers to tape when they are full. • Parent bptm **runs** /usr/openv/netbackup /bin/backup_notify on backup completion
Backup Server Class, Image and Media Database Services	Master Server database daemon: bpdbm	• bpdbm **started by** initbprd during OS boot on master server only. • In this example, bpdbm **called by** bpsched (1), bpbrm(3,9), bptm(5a, 7c). • bpdbm daemon process always running on master server - not on slave servers. • bpdbm database can be accessed locally or across network using known socket for bpdbm • bpdbm creates unique backup identifier based on client name, time of day.	2) bpdbm provides class data to bpsched to build worklist. 3) bpdbm provides configuration information to bpbrm 5a) bpdbm provides bptm with information about Backup Media database for comparison with Media Manager database. 7c) bpdbm stores Media used information into Backup Media database for bptm 9) bpdbm stores Image file list catalog for bpbrm
Backup Media Manager	bptm bpdbm	• Backup Media management handled by **calls from** bptm (5a) to bpdbm shown above.	7c) Parent bptm **calls** bpdbm to save media used data to media database. Updates keyword search for user-directed backups.
Backup Image catalog	bpbkar bpbrm bpdbm	• Image catalog handled by calls from the client bpbkar to server bpbrm and then from bpbrm to bpdbm.	8) bpbkar **writes** image catalog data to bpbrm on server 9) bpbrm **calls** bpdbm via socket to store image file list catalog.
Backup Client	bpcd - UNIX client job proxy	4) bpcd **started by** bpbrm (via inetd) to start backup jobs. Acts as proxy for bpbrm • bpcd can perform other chores, e.g. sends mail for bpbrm • bpcd returns exit status to bpbrm via socket **Activity Log:** /usr/openv/netbackup/logs/bpcd on client for client oriented commands	4) bpcd **starts** bpbkar on client for backup
	bpbkar - UNIX	4) bpbkar **started by** bpbrm (via bpcd)	4) bpbkar **Reads** file / directory list from bpbrm

Technical Example #1		Standard Duplication/Vault	
Service	**Components**	**Incoming Program / Data Flow**	**Outgoing Program / Data Flow**
	client data transfer	• bpbkar **receives** backup class options, file list, etc. from bpbrm. This information was originated by bpsched at step #1. • bpbkar **exit** status is sent to bpsched via socket **Activity Log:** /usr/openv/netbackup/logs/bpbkar on client	4) bpbkar determines NFS mounts and adds or ignores depending on class. 4) bpbkar **compresses** client data if necessary 6) bpbkar **writes** client data to bptm/bpdm on server
Media Manager	vmd ltid	5a) Media Manager vmd called by server tape manager bptm for scratch tape. 5d) Media manager ltid called by server tape manager bptm to mount tape. • See Media Manager functional design for detailed information	5c) Media Manager provides new media id for backup. Uses pool for determining which media is valid 5e) Media Manager ltid mounts tape.

Operational Procedures

This table summarizes Operational Procedures for Vault Management. The document "Bpvault - Operational Overview" provides more detailed information on day to day procedures. The document "Bpvault - Storage Administration" provides more detailed information on installation, configuration and troubleshooting:

Vault Management	Operational Procedures	
Service	**Operational Procedure**	**Staff Responsibility**
Vault Configuration	a) Review backup procedures; determine duplication capacity needed b) Assign appropriate server to run duplications; determine appropriate window for running duplication c) Edit dup_param file d) Review duplication windows for performance, throughput. e) Customize bpvault.all for email notification, report printing, etc.	1) Determine need for basic levels of duplication service on a per class basis. 2) Ensure sufficient hardware, software, network capacity available for duplication of backup images.
Vault Duplication	a) Setup cron to run bpvault.all duplication script	1) Start duplication job on time, daily and/or weekend
Vault Monitoring	a) Monitor output files to determine progress b) Setup links between log file and monitoring system for email and/or paging notification	1) Ensure duplication jobs complete successfully 2) Ensure errors reported to Event Management
Vault Report	a) Compare report output with ejected tapes, returned tapes	1) Ensure accuracy of vault process
Vault Media Management	a) Check duplication volume pool for sufficient media	1) Ensure sufficient media available for duplication
Backup Media Manager	a) Use Media Manager to manage media. b) Manually expire/freeze tapes when needed for retrieval from vault.	1)
Backup Image Catalog	a) Setup schedule for backup of image catalog. b) Ensure specific tapes available to store catalog.	1) Ensure backup catalog is safely copied.
Event Management Interface	a) Setup appropriate Event response procedures	1) Same as for normal Backup
Incident Management Interface	a) Setup procedure for passing storage events to Help Desk	1) Same as for normal Backup
Backup Reports	a) Run regular reports to ensure proper images are duplicated	1) Review production duplication cycle for thoroughness.
Duplication Capacity review	a) Determine capacity planning cycle, including reaction time, costing factors, and new requirements	1) Assist production support over time on determining system, robotic, network utilization rates and effective valuation; e.g., disk capacity. 2) Assist in new requirements and performance related additions to the system infrastructure.
Recovery Review	a) Run regular tests to ensure recovery of essential data from off-site storage	1) Ensure knowledge of appropriate procedures for restoring duplicated images. 2) Ensure sufficient knowledge to restore database catalog, backup

Vault Management	Operational Procedures	
Service	**Operational Procedure**	**Staff Responsibility**
		software, etc. in case of disaster on Netbackup server(s)
Media Manager	a) Determine media requirements and setup initial Media Poolfor duplication b) Monitor Media Pool usage over time. c) Configure Master/Slave media management	1) Ensure sufficient media is available for duplicates to run.